

MRSD Project

Tiffany May

Team G / Robographer

Teammates: Sida Wang, Rohit Dashrathi, Gauri
Gandhi, Jimit Gandhi

ILR #5

November 25, 2015

Individual progress

Primitive connection between Arduino and Matlab/Eclipse(C++ IDE):

We are trying to connect Matlab with Arduino motor code at the very first time. It works well on Windows but doesn't work on Linux. We keep getting different kinds of errors like "Open failed: Port: /dev/ttyACM0 is not available. ", permission problems and asking us to check if there's another Arduino object connected to the laptop.

After spending hours on it we start rethinking if it was worth it or we should just switch to C++ and Arduino connection. We chose the later one since some kind of the errors take time to fix. And those problems wouldn't be inherited if we switch to C++ directly.

Under Linux hardware is treated as a file located in the /dev folder and we open the relative folder to read and write to the hardware. In this case, Arduino board is also handled in the same way. For Arduino Uno, the folder will be /dev/ttyACM0.

Now, we will have two parts of code. C++ code will send the data to Arduino Uno. And Arduino Uno code will received the data.

In C++ code part, we first make a pointer *file pointed to the address "/dev/ttyACM0" and try to write random data {10, 5, 13} into Arduino Uno by fopen and fprintf. It could separate the numbers if it sensed a comma. We subtract 48 from the ascii value of the character, multiply it with its place value raised to 10 $num = num + (buff[x] - 48) * pow(10, j - x)$. But the numbers we get were not correct. Sida fixed it by changing the algorithm. There's still a small bug when counting to 100.

Control the motors with Arduino and Intraface:

With Sasanka's help(Ph.D from Kaitia's lab), we use while (Serial.available() > 0), the while function will continuously reading as long as there's data waiting to be read. We used space to separate reading for different motors and comma to separate the different reading numbers for a single motor. Function parseInt() returns the first valid (long) integer number from the serial buffer, it helps to separate the data by commas and function constrain() restricted the data in an acceptable range. Most important of all, Serial.println(panAngle, DEC) directly changed the data we need from ASCII to DEC values.

For accuracy and flexibility when controlling the position of motors, we also set the unit of delay time from ns to ms

As in the pan-tilt part and Arduino part, we tried to separated the reading window into three different parts. And if the coordinate of nose exceed the center frame, the

motor will move accordingly to keep the nose in the center. But later on we use a different strategy: proportional controller to improve the sensibility. The function introduced some problems and has not been realized yet.

Challenges

1. The connection part. We were trying to add the java.opts file to /usr/local/MATLAB/R2014b/matlab/bin/glnxa64/ and solve the writing permission problem. But the problem is now ignored.
2. ANSI transfer problem. The numbers we getting from C++ didn't transferred correctly to Arduino. Now problem was solved by using ASCII reading function from Arduino as mentioned.
3. The reading is right. But the motors don't moving accordingly after adding the proportional controller. The servo motors are vibrating but not rotating to the position we wished.

Teamwork

Sida did both the primitive and the advanced parts of connection. She connected the Intraface to Arduino, using the interested point from nose to represent the center of the whole face.

Gauri successes in reading April Tag under ROS, now she can get the id number based from the histogram of April Tag. She could also read the distance from April Tag and the camera.

Rohit implemented the elevation unit of camera set from the CAD he showed last time. The part is build from a single aluminum bar and holders created in 3D printer. There's also a screw works that we can use to adjust the height of the holder to fit different people.

Jimit figured out how to transport images from on laptop to another. He also tried to install Intraface and make it work under ROS.

Future Plan

Installing Intraface and making Intraface work on ROS.

Integration of April Tag detection, DC motors on Turtlebots, Intraface and servo motors.

